

4-14-00

A

jc520 U.S. PTO  
04/12/00

jc525 U.S. PTO  
09/548257  
04/12/00

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Inventorship.....Balaz et al.  
Applicant.....Microsoft Corporation  
Attorney's Docket No. ....MS1-467US  
Title: VPN Enrollment Protocol Gateway

TRANSMITTAL LETTER AND CERTIFICATE OF MAILING

To: Commissioner of Patents and Trademarks,  
Washington, D.C. 20231

From: Allan T. Sponseller (Tel. 509-324-9256; Fax 509-323-8979)  
Lee & Hayes, PLLC  
421 W. Riverside Avenue, Suite 500  
Spokane, WA 99201

The following enumerated items accompany this transmittal letter and are being submitted for the matter identified in the above caption.

- 1. Specification--title page, plus 45 pages, including 56 claims and Abstract
- 2. Transmittal letter including Certificate of Express Mailing
- 3. 11 Sheets Formal Drawings (Figs. 1-12)
- 4. Return Post Card

Large Entity Status ☒ Small Entity Status ☐

Date: April 12, 2000 By: [Signature]  
Allan T. Sponseller  
Reg. No. 38,318

CERTIFICATE OF MAILING

I hereby certify that the items listed above as enclosed are being deposited with the U.S. Postal Service as either first class mail, or Express Mail if the blank for Express Mail No. is completed below, in an envelope addressed to The Commissioner of Patents and Trademarks, Washington, D.C. 20231, on the below-indicated date. Any Express Mail No. has also been marked on the listed items.

EL580803807

Express Mail No. (if applicable) \_\_\_\_\_

Date: 4.12.00 By: [Signature]  
Lori A. Vierra

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION FOR LETTERS PATENT

**VPN Enrollment Protocol Gateway**

Inventor(s):  
Rudolph Balaz  
Victor W. Heller  
Xiaohong Su  
Keith R. Vogel

ATTORNEY'S DOCKET NO. MS1-467US

## **TECHNICAL FIELD**

This invention relates to secure communications, and more particularly to a protocol gateway allowing routers operating in accordance with one protocol to obtain and maintain certificates for a virtual private network (VPN) from a certificate authority operating in accordance with another protocol.

## **BACKGROUND OF THE INVENTION**

Computer technology is continually advancing, resulting in continually evolving uses for computers. One such use is communicating with other computers over a network, such as the Internet, to obtain or exchange information, purchase or sell goods or services, etc. One particular type of communication that can be established is referred to as a "virtual private network" or "VPN". In a VPN, portions of a network (such as the Internet) are used to establish secure communications from one computer to another via multiple different routers in the network. The VPN allows users to use the larger network (e.g., the Internet) to connect to another computer as if they were part of a dedicated secure network.

In order to operate as part of a VPN, a router enrolls for a VPN certificate via a certificate authority (CA). This VPN certificate is then provided to other routers that are part of the VPN and is used to authenticate the router and may also be used to securely communicate with the other routers. However, different protocols for enrolling for VPN certificates have arisen, many of which are incompatible with one another. For example, many routers available from Cisco Systems, Inc. of San Jose, California use a proprietary protocol called Simple Certificate Enrollment Protocol (SCEP) for obtaining VPN certificates, while many certificate authorities available from Microsoft Corporation of Redmond,

1 Washington use an incompatible enrollment protocol based on Public-Key  
2 Cryptography Standard (PKCS) #10 and PKCS #7. Thus, a router using SCEP  
3 would not be able to enroll for a VPN certificate from a CA using PKCS #10 and  
4 PKCS #7.

5 Additionally, many routers and CAs are already manufactured and in use  
6 that operate based on such incompatible protocols. Therefore, re-designing such  
7 routers or CAs to be compatible with one another would require the replacement  
8 of many such pre-existing devices. Thus, it would be beneficial to provide a  
9 solution that allows routers and CAs (including pre-existing routers and CAs)  
10 operating based on incompatible protocols to communicate with one another for  
11 VPN certificate enrollment.

12 The VPN enrollment protocol gateway described below addresses these and  
13 other disadvantages.

## 14 15 **SUMMARY OF THE INVENTION**

16 A virtual private network (VPN) enrollment protocol gateway is described  
17 herein. The protocol gateway allows routers operating in accordance with one  
18 protocol to obtain and maintain certificates for a VPN from a certificate authority  
19 operating in accordance with another protocol.

20 According to one aspect, the VPN enrollment protocol gateway is  
21 implemented as a registration authority that operates as an intermediary between  
22 the router and the certificate authority. As a registration authority, the gateway is  
23 trusted by the certificate authority. The router communicates with the registration  
24 authority as if it were the certificate authority, not realizing that it is  
25 communicating with an intermediary.

1 According to another aspect, the protocol gateway receives a router  
2 enrollment request from the router. The protocol gateway decrypts the request,  
3 adds an alternative subject name to the request, digitally signs the request, and  
4 forwards the signed request to the certificate authority. The certificate authority  
5 determines whether to trust the source of the request (the protocol gateway), and  
6 proceeds to respond with the requested certificate if it verifies that the gateway can  
7 be trusted. The gateway receives the requested certificate, encrypts and digitally  
8 signs a response including the certificate, and returns the signed and encrypted  
9 response to the router.

10 According to another aspect, the certificate authority may not be able to  
11 immediately issue a certificate, in which case it issues a pending response. The  
12 registration authority maintains a mapping of a router transaction ID (identifier)  
13 received from the router and a pending response ID received from the certificate  
14 authority. This mapping allows subsequent requests from the router with the same  
15 transaction ID (e.g., querying whether the certificate has been issued yet) to be  
16 properly matched to a request previously submitted to the certificate authority for  
17 which a pending response was issued. The registration authority also maintains a  
18 mapping of a hash value of the request received from the router to the pending  
19 response for that request. This mapping allows the registration authority to  
20 determine when a request is resubmitted by the router (e.g., in the event the router  
21 never receives a pending response returned to it by the registration authority).

22 According to another aspect, the protocol gateway receives a get certificate  
23 revocation list from the router. The protocol gateway decrypts the request and  
24 extracts from the request the certificate serial number of the signing certificate of  
25 the request. The protocol gateway then submits a Get Certificate by Serial

1 Number request to the certificate authority, which returns to the protocol gateway  
2 the certificate corresponding to the serial number. The protocol gateway extracts a  
3 certificate revocation list distribution point from the response, and obtains the  
4 certificate revocation list from the distribution point. The protocol gateway then  
5 generates a response including the certificate revocation list, encrypts and signs  
6 the response, and returns the response to the router.

7 According to another aspect, the protocol gateway receives a get certificate  
8 request from the router. The protocol gateway decrypts the request and extracts  
9 from the request the certificate serial number of the signing certificate of the  
10 request. The protocol gateway then submits a Get Certificate by Serial Number  
11 request to the certificate authority, which returns to the protocol gateway the  
12 certificate corresponding to the serial number. The protocol gateway then encrypts  
13 and signs a response including the certificate, and returns the response to the  
14 router.

15 According to another aspect, the protocol gateway receives a get certificate  
16 authority certificate request from the router. The protocol gateway generates a  
17 response message including the signing certificate of the registration authority as  
18 well as the encryption certificate of the registration authority, and returns the  
19 response message to the router.

20 According to another aspect, the protocol gateway maintains a record of  
21 passwords handed out to a router. A router obtains a password by communicating  
22 with the protocol gateway (or another device trusted by the protocol gateway) via  
23 an authenticatable mechanism (e.g., SSL (Secure Sockets Layer)). A password is  
24 returned to the router, which can then use this password for a request submitted to  
25

1 the protocol gateway. If the password presented by the router is in the router's  
2 record, then the request is processed; otherwise, the request is rejected.

### 3 4 **BRIEF DESCRIPTION OF THE DRAWINGS**

5 The present invention is illustrated by way of example and not limitation in  
6 the figures of the accompanying drawings. The same numbers are used  
7 throughout the figures to reference like components and/or features.

8 Fig. 1 shows a virtual private network environment with an enrollment  
9 protocol gateway in accordance with certain embodiments of the invention.

10 Fig. 2 shows a general example of a computer that can be used in  
11 accordance with certain embodiments of the invention.

12 Fig. 3 is a block diagram illustrating a registration authority operating as a  
13 protocol gateway between a router and a certificate authority in accordance with  
14 certain embodiments of the invention.

15 Fig. 4 shows an exemplary transaction ID table in accordance with certain  
16 embodiments of the invention.

17 Fig. 5 shows an exemplary request hash table in accordance with certain  
18 embodiments of the invention.

19 Fig. 6 shows an exemplary password table in accordance with certain  
20 embodiments of the invention.

21 Figs. 7a and 7b are a flowchart illustrating an exemplary process for  
22 handling a router enrollment request in accordance with certain embodiments of  
23 the invention.

24 Fig. 8 is a flowchart illustrating an exemplary process for handling pending  
25 responses in accordance with certain embodiments of the invention.

1 Fig. 9 is a flowchart illustrating an exemplary process for handling a Get  
2 Certificate Revocation List request in accordance with certain embodiments of the  
3 invention.

4 Fig. 10 is a flowchart illustrating an exemplary process for handling a Get  
5 Certificate request in accordance with certain embodiments of the invention.

6 Fig. 11 is a flowchart illustrating an exemplary process for handling a Get  
7 Certificate Authority Certificate request in accordance with certain embodiments  
8 of the invention.

9 Fig. 12 is a flowchart illustrating an exemplary process for distributing and  
10 verifying passwords in accordance with certain embodiments of the invention.

## 11 DETAILED DESCRIPTION

12 The discussion herein assumes that the reader is familiar with cryptography.  
13 For a basic introduction of cryptography, the reader is directed to a text written by  
14 Bruce Schneier and entitled "Applied Cryptography: Protocols, Algorithms, and  
15 Source Code in C," published by John Wiley & Sons with copyright 1994 (or  
16 second edition with copyright 1996).  
17

18 In the discussion below, embodiments of the invention will be described in  
19 the general context of computer-executable instructions, such as program modules,  
20 being executed by one or more conventional personal computers. Generally,  
21 program modules include routines, programs, objects, components, data structures,  
22 etc. that perform particular tasks or implement particular abstract data types.  
23 Moreover, those skilled in the art will appreciate that various embodiments of the  
24 invention may be practiced with other computer system configurations, including  
25 hand-held devices, multiprocessor systems, microprocessor-based or



1 programmable consumer electronics, network PCs, minicomputers, mainframe  
2 computers, and the like. In a distributed computer environment, program modules  
3 may be located in both local and remote memory storage devices.

4 Alternatively, embodiments of the invention can be implemented in  
5 hardware or a combination of hardware, software, and/or firmware. For example,  
6 all or part of the invention can be implemented in one or more application specific  
7 integrated circuits (ASICs).

8 Fig. 1 shows a virtual private network environment with an enrollment  
9 protocol gateway in accordance with certain embodiments of the invention.  
10 Generally, one or more client computers 102 can communicate with one or more  
11 server computers 104 via a public network supporting a conventional virtual  
12 private network (VPN) 106. Server computers 104 can be coupled directly to the  
13 network supporting VPN 106, or alternatively can be coupled to the network  
14 supporting VPN 106 via another network, such as local area network (LAN) 108.

15 VPN 106 includes one or more routers 110, 112, and 114 through which  
16 data is passed between client 102 and server 104. Routers 110 – 114 are part of a  
17 public network, such as the Internet. Routers that are part of other types of  
18 networks may also be included in VPN 106, such as routers from a LAN or a  
19 private wide-area network.

20 Additionally, other networks may be involved in the communication  
21 between client 102 and server 104. By way of example, client 102 may connect to  
22 the public network supporting VPN 106 via a conventional modem and a Public  
23 Switched Telephone Network (PSTN), via a conventional cable modem and cable  
24 lines, etc.

1 Routers 110 – 114 can communicate with one another, as well as  
2 registration authority 118, via any of a wide variety of conventional  
3 communications protocols. In one implementation, routers 110 – 114  
4 communicate with one another and registration authority 118 using the Hypertext  
5 Transfer Protocol (HTTP).

6 Each of the routers 110 – 114 receives data from one of the other routers  
7 110 – 114 or alternatively from another component (e.g., a public network access  
8 provider, such as an Internet Service Provider (ISP); client computer 102; etc.).  
9 The data is then securely passed on to another of the routers 110 – 114 or other  
10 components.

11 In order for data to be transmitted among routers 110 – 114, a certificate-  
12 based authentication scheme is employed. In such an authentication scheme, each  
13 router 110 – 114 is assigned a unique certificate that it can use to authenticate  
14 itself to other routers or other computing devices (e.g., an ISP, a bridge or  
15 gateway, etc.). Additionally, these other computing devices may be part of VPN  
16 106 and may similarly be assigned unique certificates that can be used for  
17 authentication. Such certificates can also optionally be used to encrypt messages  
18 between routers and/or other computing devices in any of a variety of  
19 conventional manners. For ease of explanation, routers are described as the  
20 devices that are obtaining and maintaining certificates for VPN 106. The  
21 establishment and operation of a VPN is well-known to those skilled in the art,  
22 and thus will not be discussed further except as it pertains to the invention.

23 The certificates used by routers 110 – 114 are assigned by a trusted  
24 certificate authority (CA) 116. The process of obtaining such a certificate is  
25 referred to as “enrollment”. In the illustrated example, routers 110 – 114 use a

1 different enrollment protocol than is used by certificate authority 116. A  
2 registration authority 118 communicates with both routers 110 – 114 and  
3 certificate authority 116 and acts as an intermediary for enrollment, translating  
4 requests and responses in one protocol to another, as discussed in more detail  
5 below.

6 Fig. 2 shows a general example of a computer 142 that can be used in  
7 accordance with certain embodiments of the invention. Computer 142 is shown as  
8 an example of a computer that can perform the functions of a client computer 102,  
9 a server computer 104, a certificate authority 116, or a registration authority 118  
10 of Fig. 1. Computer 142 includes one or more processors or processing units 144,  
11 a system memory 146, and a bus 148 that couples various system components  
12 including the system memory 146 to processors 144.

13 The bus 148 represents one or more of any of several types of bus  
14 structures, including a memory bus or memory controller, a peripheral bus, an  
15 accelerated graphics port, and a processor or local bus using any of a variety of  
16 bus architectures. The system memory includes read only memory (ROM) 150  
17 and random access memory (RAM) 152. A basic input/output system (BIOS) 154,  
18 containing the basic routines that help to transfer information between elements  
19 within computer 142, such as during start-up, is stored in ROM 150. Computer  
20 142 further includes a hard disk drive 156 for reading from and writing to a hard  
21 disk, not shown, connected to bus 148 via a hard disk driver interface 157 (e.g., a  
22 SCSI, ATA, or other type of interface); a magnetic disk drive 158 for reading from  
23 and writing to a removable magnetic disk 160, connected to bus 148 via a  
24 magnetic disk drive interface 161; and an optical disk drive 162 for reading from  
25 or writing to a removable optical disk 164 such as a CD ROM, DVD, or other

1 optical media, connected to bus 148 via an optical drive interface 165. The drives  
2 and their associated computer-readable media provide nonvolatile storage of  
3 computer readable instructions, data structures, program modules and other data  
4 for computer 142. Although the exemplary environment described herein employs  
5 a hard disk, a removable magnetic disk 160 and a removable optical disk 164, it  
6 should be appreciated by those skilled in the art that other types of computer  
7 readable media which can store data that is accessible by a computer, such as  
8 magnetic cassettes, flash memory cards, digital video disks, random access  
9 memories (RAMs) read only memories (ROM), and the like, may also be used in  
10 the exemplary operating environment.

11 A number of program modules may be stored on the hard disk, magnetic  
12 disk 160, optical disk 164, ROM 150, or RAM 152, including an operating system  
13 170, one or more application programs 172, other program modules 174, and  
14 program data 176. Operating system 170 can be any of a variety of operating  
15 systems, such as any of the "Windows" family of operating systems available from  
16 Microsoft Corporation of Redmond, Washington. A user may enter commands  
17 and information into computer 142 through input devices such as keyboard 178  
18 and pointing device 180. Other input devices (not shown) may include a  
19 microphone, joystick, game pad, satellite dish, scanner, or the like. These and  
20 other input devices are connected to the processing unit 144 through an interface  
21 168 (e.g., a serial port interface) that is coupled to the system bus. A monitor 184  
22 or other type of display device is also connected to the system bus 148 via an  
23 interface, such as a video adapter 186. In addition to the monitor, personal  
24 computers typically include other peripheral output devices (not shown) such as  
25 speakers and printers.

1 Computer 142 can operate in a networked environment using logical  
2 connections to one or more remote computers, such as a remote computer 188.  
3 The remote computer 188 may be another personal computer, a server, a router, a  
4 network PC, a peer device or other common network node, and typically includes  
5 many or all of the elements described above relative to computer 142, although  
6 only a memory storage device 190 has been illustrated in Fig. 2. The logical  
7 connections depicted in Fig. 2 include a local area network (LAN) 192 and a wide  
8 area network (WAN) 194. Such networking environments are commonplace in  
9 offices, enterprise-wide computer networks, intranets, and the Internet. In the  
10 described embodiment of the invention, remote computer 188 executes an Internet  
11 Web browser program such as the "Internet Explorer" Web browser manufactured  
12 and distributed by Microsoft Corporation of Redmond, Washington.

13 When used in a LAN networking environment, computer 142 is connected  
14 to the local network 192 through a network interface or adapter 196. When used  
15 in a WAN networking environment, computer 142 typically includes a modem 198  
16 or other means for establishing communications over the wide area network 194,  
17 such as the Internet. The modem 198, which may be internal or external, is  
18 connected to the system bus 148 via a serial port interface 168. In a networked  
19 environment, program modules depicted relative to the personal computer 142, or  
20 portions thereof, may be stored in the remote memory storage device. It will be  
21 appreciated that the network connections shown are exemplary and other means of  
22 establishing a communications link between the computers may be used.

23 Generally, the data processors of computer 142 are programmed by means  
24 of instructions stored at different times in the various computer-readable storage  
25 media of the computer. Programs and operating systems are typically distributed,

1 for example, on floppy disks or CD-ROMs. From there, they are installed or  
2 loaded into the secondary memory of a computer. At execution, they are loaded at  
3 least partially into the computer's primary electronic memory. The invention  
4 described herein includes these and other various types of computer-readable  
5 storage media when such media contain instructions or programs for implementing  
6 the steps described below in conjunction with a microprocessor or other data  
7 processor. The invention also includes the computer itself when programmed  
8 according to the methods and techniques described below. Furthermore, certain  
9 sub-components of the computer may be programmed to perform the functions  
10 and steps described herein. The invention includes such sub-components when  
11 they are programmed as described. In addition, the invention described herein  
12 includes data structures, described herein, as embodied on various types of  
13 memory media.

14 For purposes of illustration, programs and other executable program  
15 components such as the operating system are illustrated herein as discrete blocks,  
16 although it is recognized that such programs and components reside at various  
17 times in different storage components of the computer, and are executed by the  
18 data processor(s) of the computer.

19 Fig. 3 is a block diagram illustrating an exemplary registration authority  
20 118 operating as a protocol gateway between a router 210 and a certificate  
21 authority 116. Router 210 can be, for example, any of routers 110 – 114 of Fig. 1.  
22 Router 210 is configured (e.g., during an installation or setup process) with the  
23 address of registration authority 118 rather than CA 116 as the certificate  
24 authority. In the illustrated example, router 210 has no other knowledge that it is  
25

1 communicating with registration authority 118 rather than certificate authority  
2 116.

3       Communication between registration authority 118 and each of router 210  
4 and certificate authority 116 can be carried out using any of a wide variety of  
5 conventional encryption and/or digital signing techniques. By way of example,  
6 using well-known public key cryptography techniques, a device obtains a private  
7 key/public key pair; the public key is made available to other devices while the  
8 private key is kept secret by the device. Another device can encrypt a message  
9 intended for this device by using a conventional encryption algorithm and this  
10 device's public key. The private key/public key pair and the encryption algorithm  
11 are chosen such that it is relatively easy to decrypt the message with the private  
12 key, but extremely difficult to decrypt the message without the private key.  
13 Similarly, a message can be digitally signed by the device using a conventional  
14 encryption algorithm and its private key. The digitally signed message can be  
15 decrypted by another device using the public key, allowing the other device to  
16 verify that the message came from that device. Alternatively, rather than applying  
17 an encryption algorithm to the message itself, the encryption algorithm may be  
18 applied to a hash value generated based on the message and a known hash  
19 function. Different public key/private key pairs can be used for encryption and  
20 digital signatures, or alternatively the same public key/private key pair can be used  
21 for both encryption and digital signatures.

22       Registration authority 118 operates as an enrollment agent for certificate  
23 authority 116, allowing routers such as router 210 to enroll for a VPN certificate  
24 from certificate authority 116 via registration authority 118. Registration authority  
25 118 obtains, from certificate authority 116, an enrollment agency signature

1 certificate (e.g., by enrolling for an "Offline IPsec" enrollment agent signature  
2 certificate) and an encryption certificate (e.g., by enrolling for an "IPsec  
3 Encryption" certificate). In the illustrated examples, these certificates are used by  
4 registration authority 118 to digitally sign data sent to both the router 210 and the  
5 certificate authority 116, and to encrypt data sent to the router 210.

6 Router 210 communicates requests 212 to registration authority 118 in  
7 accordance with the protocol supported by router 210. In the illustrated example,  
8 router 210 supports the protocol SCEP. Different types of requests 212 can be  
9 transmitted to registration authority 118. In one implementation, registration  
10 authority 118 operates as a protocol gateway for the following types of requests:  
11 router enrollment, get certificate revocation list (CRL), get certificate, get  
12 certificate authority (CA) certificate, and password registration. The specific  
13 manner in which each of these requests is handled by registration authority 118 is  
14 discussed in more detail below.

15 Upon receipt of an SCEP request 212, registration authority 118 converts  
16 the request into an appropriate format for certificate authority 116. The converted  
17 request is then digitally signed by registration authority 118 and the signed request  
18 214 is transmitted to certificate authority 116. Certificate authority 116, receiving  
19 a request in its own protocol (using PKCS #7 and PKCS #10), responds to the  
20 request and issues a CA response 216. Registration authority 118 receives the  
21 response 216, converts the response to the appropriate SCEP format for router  
22 210, and transmits an SCEP response 218 to router 210. Alternatively, for some  
23 requests registration authority 118 may generate the response 218 without  
24 forwarding a signed request 214 to certificate authority 116.  
25



1 Registration authority 118 includes a protocol converter 220. Protocol  
2 converter 220 receives messages from router 210 and converts them as necessary  
3 to the proper protocol for certificate authority 116, and similarly receives  
4 messages from certificate authority 116 and converts them to the proper protocol  
5 for router 210. The manner in which protocol converter 220 operates is dependent  
6 on the particular protocols being used by router 210 and certificate authority 116.

7 In one implementation, registration authority 118 operates in accordance  
8 with the Internet X.509 Public Key Infrastructure Certificate and CRL Profile  
9 (Network Working Group Request for Comments 2459, January 1999).  
10 Alternatively, other implementations may operate in accordance with other  
11 standards.

12 Registration authority 118 also includes a transaction ID table 222, a  
13 request hash table 224, and a password table 226. Tables 222 – 226 are used by  
14 registration authority 118 to maintain information regarding requests 212 and  
15 responses 216 in order to conform with the protocols of router 210 and certificate  
16 authority 116.

17 Fig. 4 shows an exemplary transaction ID table in accordance with certain  
18 embodiments of the invention. Transaction ID table 222 maintains a mapping of  
19 router transaction IDs 228 to CA request IDs 230. A router transaction ID 228 is  
20 received by registration authority 118 from router 210 as part of each router  
21 enrollment message. Similarly, when certificate authority 116 returns a pending  
22 response to registration authority 118, the pending response includes a CA request  
23 ID 230 (also referred to as a "token"). Transaction ID table 222 allows registration  
24 authority 118 to query certificate authority 116 for the correct certificate in  
25

1 response to subsequent requests from router 210 for the certificate the pending  
2 response was issued for, as discussed in more detail below.

3 Each entry in transaction ID table 222 is removed from table 222 after a  
4 period of time. In one implementation, each entry in table 222 is kept in table 222  
5 for one week and then removed. This period of time can optionally be  
6 configurable by a user or administrator.

7 Fig. 5 shows an exemplary request hash table in accordance with certain  
8 embodiments of the invention. Request hash table 224 maintains a mapping of  
9 certificate authority request IDs 232 to hash values of the requests 234. The hash  
10 value of a request is generated using any of a variety of conventional hashing  
11 functions, such as MD5 (Message Digest 5). A hash function is a mathematical  
12 function that, given input data (e.g., the request) generates a unique output hash  
13 value based on the input data. Thus, the hash value uniquely identifies a request  
14 but requires less storage space than maintaining all of the request. Alternatively,  
15 table 224 could maintain the actual request rather than hash values of the request.

16 Request hash table 224 allows registration authority 118 to "remember"  
17 router requests. For example, a pending response may be issued by registration  
18 authority 118 to router 210, as discussed in more detail below. If a failure or  
19 problem occurs during the transmission (e.g., a network failure), then the pending  
20 response may not be received by router 210. If router 210 never receives the  
21 response, router 210 will re-issue the same request. By maintaining table 224,  
22 registration authority 118 can determine when a received request is a re-issued  
23 request, and need not submit another request for another new certificate to  
24 certificate authority 116.

1 Each entry in request hash table 224 is removed from table 224 after a  
2 period of time. In one implementation, each entry in table 224 is kept in table 224  
3 for twenty minutes and then removed. This period of time can optionally be  
4 configurable by a user or administrator.

5 Fig. 6 shows an exemplary password table in accordance with certain  
6 embodiments of the invention. Password table 226 maintains passwords 236 that  
7 are issued to router 210 in a secure manner. Such passwords can subsequently be  
8 used by router 210 to obtain a certificate, providing verification of the identity of  
9 router 210.

10 Each password in password table 226 is removed from table 226 after a  
11 period of time. In one implementation, each password in table 226 is kept in table  
12 226 for sixty minutes and then removed. This period of time can optionally be  
13 configurable by an administrator.

14 Returning to Fig. 3, in the illustrated example registration authority 118 is a  
15 dynamically linked library (DLL) referred to as the "MSCEP" DLL.  
16 Alternatively, registration authority 118 may include a DLL referred to as the  
17 "MSCEP" DLL. Registration authority 118 includes a response module 238 that  
18 generates responses for certain requests from router 210 that do not require  
19 forwarding to certificate authority 116. The operation of response module 238 is  
20 discussed in more detail below.

21 Registration authority 118 further hosts a web site 240. Alternatively,  
22 registration authority 118 may have a secure communication link to a server  
23 hosting web site 240, thereby allowing data to be securely passed between the  
24 server and registration authority 118, or registration authority 118 may be software  
25 and/or firmware being executed by a server that also hosts web site 240. Web site

1 240 allows passwords to be securely issued to router 210 and stored in password  
2 table 226, as discussed in more detail below.

### 3 4 **Router Enrollment Request**

5 Figs. 7a and 7b are a flowchart illustrating an exemplary process for  
6 handling a router enrollment request in accordance with certain embodiments of  
7 the invention. Acts on the left-hand side of Figs. 7a and 7b are implemented by  
8 registration authority 118 of Fig. 3, while acts on the right-hand side are  
9 implemented by certificate authority 116. The process of Figs. 7a and 7b may be  
10 performed in software, firmware, hardware, or a combination thereof. Figs. 7a  
11 and 7b are described with additional reference to components in Fig. 3.

12 To participate in a VPN, router 210 enrolls for a certificate from certificate  
13 authority 116. Router 210 enrolls for a certificate by sending, as SCEP request  
14 212, a router enrollment message (e.g., a SCEP PKCSReq message) to registration  
15 authority 118. The router enrollment message includes a certificate enrollment  
16 request in accordance with the Public-Key Cryptography Standards (PKCS) #10  
17 standard. The certificate enrollment request is further encrypted (e.g., using the  
18 public key of registration authority 118) and then digitally signed by router 210 in  
19 accordance with the Public-Key Cryptography Standards (PKCS) #7 standard.  
20 Additional information regarding PKCS #7 and PKCS #10 is available from RSA  
21 Data Security, Inc. of Bedford, MA. It should be noted that, although requests  
22 from router 210 use PKCS #7 and PKCS #10, certain information needed by  
23 certificate authority 116 is not included in the requests. Registration authority 118  
24 resolves this problem, adding information when necessary.

1 Registration authority 118 receives, as the router enrollment message, this  
2 encrypted and digitally signed request (act 242). Upon receipt of the enrollment  
3 message, registration authority 118 verifies the signature of the router enrollment  
4 message (act 244). If the signature is not verified then the message is ignored (act  
5 246). Alternatively, an indication of failure could be returned to router 210.

6 If the signature is verified, then registration authority 118 decrypts the  
7 router enrollment message (e.g., using the private key of registration authority  
8 118) and extracts the certificate enrollment request from the message (act 248).  
9 Registration authority 118 uses the certificate enrollment request to generate a  
10 request to the CA for an enrollment certificate in a format expected by certificate  
11 authority 116 (act 250).

12 Router 210 needs a certificate with a subject alternative names extension  
13 (SubjectAltName). However, router 210 does not specifically request the  
14 SubjectAltName extension, and certificate authority 116 does not automatically  
15 add the extension. Registration authority 118 resolves this issue by adding, to the  
16 message it transmits to certificate authority 116, the SubjectAltName extension in  
17 the request.

18 The PKCS #7 message, including both the subject alternative names  
19 extension and the certificate enrollment request extracted from the router  
20 enrollment message, is digitally signed by registration authority 118 (act 252).  
21 This signed message is then transmitted to certificate authority 116 as a CA  
22 request (act 254). Note that the CA request thus includes a PKCS #7 message that  
23 is signed by registration authority 118, which in turn includes a certificate  
24 enrollment request that is signed by router 210.  
25

1 Certificate authority 116 receives the CA request from registration authority  
2 118 (act 256) and determines, based on the content of the CA request, whether to  
3 issue the requested certificate (act 258). The manner in which certificate authority  
4 116 determines whether to issue the requested certificate can vary. In one  
5 implementation, certificate authority 116 determines whether to issue a certificate  
6 based on whether the certificate of the registration authority 118 can be validated  
7 up to a trusted valid root and whether the certificate of registration authority 118  
8 includes an extended key usage indicating that registration authority 118 can be a  
9 registration authority (and thus operate as an enrollment agent). If both of these  
10 conditions are satisfied, then a certificate is issued. Otherwise, the certificate is  
11 not issued. Additionally, certificate authority 116 may require that the certificate  
12 of registration authority 118 have been issued directly by a certificate authority  
13 (that is, no intermediate certificates in the chain from the registration authority  
14 certificate to the certificate authority certificate).

15 If certificate authority 116 determines it will not issue a certificate, then  
16 certificate authority 116 generates a CA response indicating failure (act 260).  
17 However, if certificate authority 116 determines it will issue a certificate, then  
18 certificate authority 116 generates the requested certificate (act 262) and then  
19 generates a CA response including the generated certificate (act 264).

20 The CA response generated by certificate authority 116 has no message  
21 content and is referred to as a "degenerated PKCS #7". The PKCS #7 message,  
22 however, allows multiple certificates to be included in a degenerated PKCS #7  
23 message. Certificate authority 116 returns the newly generated certificate as part  
24 of the degenerated PKCS #7 message. Additionally, the entire certificate chain  
25

1 from the generated certificate up to a root certificate may optionally be included in  
2 the degenerated PKCS #7 message.

3 Certificate authority 116 then transmits the CA response (indicating either  
4 failure or with the generated certificate) to registration authority 118 (act 266).  
5 Registration authority 118 receives the CA response (act 268) and checks whether  
6 the CA response includes a certificate (act 270). If no certificate is included, then  
7 registration authority 118 generates an SCEP response message indicating failure  
8 (act 272). However, if such a certificate is included, then registration authority  
9 118 extracts the certificate (act 274) and generates an SCEP response including the  
10 certificate (act 276). In the illustrated example, registration authority 118 extracts  
11 only the certificate generated by certificate authority 116; the additional certificate  
12 chain (if included) is not used by registration authority 118. Alternatively, the  
13 entire certificate chain could be included if router 210 desired (or at least could  
14 handle) the chain.

15 Registration authority 118 then encrypts the SCEP response (act 278) and  
16 digitally signs the encrypted response (act 280). The encrypted and signed  
17 response is then transmitted to router 210 (act 282), which in turn can verify the  
18 signature and decrypt the response to extract the certificate generated by certificate  
19 authority 116.

### 20 21 **Pending Response Handling**

22 In some situations, certificate authority 116 may not immediately issue a  
23 CA response with either a certificate or an indication that no certificate will be  
24 issued. For example, certificate authority 116 may wait for an administrator to  
25

1 approve the issuing of the certificate. In such situations, certificate authority 116  
2 issues a CA pending response from certificate authority 116.

3 Fig. 8 is a flowchart illustrating an exemplary process for handling pending  
4 responses in accordance with certain embodiments of the invention. The process  
5 of Fig. 8 is implemented by registration authority 118 of Fig. 3, and may be  
6 performed in software, firmware, hardware, or a combination thereof. Fig. 8 is  
7 described with additional reference to components in Figs. 3 - 7b.

8 Registration authority 118 receives the CA pending response from  
9 certificate authority 116 (act 302). Upon receipt of the CA pending response,  
10 registration authority 118 adds entries to its transaction ID table 222 (act 304) and  
11 its request hash table 224 (act 306). Registration authority 118 also generates an  
12 encrypted and digitally signed SCEP pending response message (act 308) and  
13 transmits the encrypted and signed message to router 210 (act 310).

14 Typically, in response to an SCEP pending response message, router 210  
15 will re-issue its request for a certificate (e.g., via a GetCertInitial message).  
16 Registration authority 118 waits until it receives an additional SCEP request for  
17 the certificate from the router 210 (act 312). Once the additional request is  
18 received, registration authority 118 accesses transaction ID table 222 to determine  
19 the appropriate CA request ID (act 314). Registration authority 118 uses the CA  
20 request ID from table 222 to generate a CA request for a certificate corresponding  
21 to the CA request ID and digitally signs the CA request (act 316). The signed CA  
22 request is then transmitted to certificate authority 116 (act 318).

23 Upon receiving the CA request, certificate authority 116 may issue another  
24 pending response to registration authority 118 or alternatively determine whether  
25 to issue the certificate (per act 258 of Fig. 7a discussed above). Upon receipt of a



1 response from certificate authority 116, registration authority 118 determines  
2 whether the response is another pending response (act 320). If the response is  
3 another pending response, the registration authority 118 returns to act 308 and  
4 generates and encrypted and signed SCEP pending response message. However, if  
5 the response is not another pending response, then registration authority 118  
6 proceeds per acts 268 – 282 of Fig. 7b to return an appropriate response to router  
7 210.

8 Use of request hash table 224 further allows registration authority 118 to  
9 gracefully recover in the event the SCEP pending response message is not  
10 received by router 210. If router 210 does not receive the pending response  
11 message, then it will resubmit its original request (e.g., an SCEP PKCSReq  
12 message). In order to avoid a duplicate request to certificate authority for the  
13 certificate, registration authority 118 generates the hash value for SCEP PKCSReq  
14 messages it receives and compares the hash value to the entries in request hash  
15 table 224. If the hash value matches an entry, then registration authority 118 uses  
16 the CA request ID from table 224 to generate a CA request for a certificate  
17 corresponding to the CA request ID (act 316), rather than generating a CA request  
18 including a certificate enrollment request (act 250 of Fig. 7a). Processing then  
19 continues as discussed above with reference to Fig. 8.

### 21 **Get Certificate Revocation List Request**

22 Returning to Fig. 3, router 210 may also send a Get Certificate Revocation  
23 List (CRL) request as SCEP request 212. The request identifies a serial number or  
24 similar identifier of a certificate for which the corresponding CRL should be  
25 retrieved. The CRL is a list identifying revoked certificates which is made

1 available by the certificate authority (typically in a public repository). The CRL  
2 can be checked to determine whether a particular serial number (typically  
3 identified in the CRL by its serial number) has been revoked. Registration  
4 authority 118 responds to such a request by obtaining the requested CRL and  
5 returning it to router 210.

6 Fig. 9 is a flowchart illustrating an exemplary process for handling a Get  
7 Certificate Revocation List request in accordance with certain embodiments of the  
8 invention. The process of Fig. 9 is implemented by registration authority 118 of  
9 Fig. 3, and may be performed in software, firmware, hardware, or a combination  
10 thereof. Fig. 9 is described with additional reference to components in Fig. 3.

11 Initially, registration authority 118 receives the Get CRL request (e.g., an  
12 SCEP GetCRL message) from router 210 (act 330). Registration authority 118  
13 decrypts the request (act 332), verifies the signature of the decrypted request (act  
14 334), and proceeds based on whether the signature is verified (act 336). If the  
15 signature cannot be successfully verified, then the message is dropped (act 338);  
16 registration authority 118 simply ignores the message. Alternatively, registration  
17 authority 118 may return an indication to router 210 that the signature could not be  
18 verified.

19 However, if the signature is successfully verified, then registration authority  
20 118 extracts the certificate serial number from the decrypted request (act 340).  
21 This serial number can be extracted by obtaining the serial number of the  
22 certificate used by router 210 to sign the Get CRL request.

23 Registration authority 118 then uses the extracted serial number to generate  
24 a Get Certificate by Serial Number request (act 342). The Get Certificate by  
25 Serial Number request is then digitally signed and transmitted to certificate

1 authority 116 (act 344), which in turn accesses its records to identify the certificate  
2 corresponding to the given serial number. This certificate is then returned by  
3 certificate authority 116 to registration authority 118 (act 346).

4 The certificate returned by certificate authority 116 includes a CRL  
5 distribution point, which is an identifier of a location (e.g., a uniform resource  
6 locator (URL)) at which the CRL corresponding to the certificate can be obtained.  
7 Upon receipt of the certificate, registration authority 118 extracts the CRL  
8 distribution point from the certificate (act 348). Registration authority 118 then  
9 accesses (e.g., via HTTP) the identified location and retrieves the CRL located  
10 there (act 350).

11 Upon obtaining the CRL, registration authority 118 generates an SCEP  
12 response message including the CRL (act 352). Registration authority 118 then  
13 encrypts and digitally signs the SCEP response message including the CRL, and  
14 returns the encrypted and signed SCEP response message to router 210 (act 354).

15 Alternatively, the Get CRL request received from router 210 (act 330) may  
16 include the certificate for which the corresponding CRL is to be obtained. In this  
17 situation, the CRL distribution point can be extracted by accessing the included  
18 certificate, thereby alleviating the need to access certificate authority 116 (acts 340  
19 – 346).

### 20 21 **Get Certificate Request**

22 Returning to Fig. 3, router 210 may also send a Get Certificate request as  
23 SCEP request 212. The request identifies a serial number of a certificate that the  
24 router would like returned to it. Router 210 may make such a request, for  
25 example, in situations where it has kept the serial number of a certificate it needs

1 but has not kept the actual certificate. Registration authority 118 responds to such  
2 a request by obtaining the requested certificate and returning it to router 210.

3 Fig. 10 is a flowchart illustrating an exemplary process for handling a Get  
4 Certificate request in accordance with certain embodiments of the invention. The  
5 process of Fig. 10 is implemented by registration authority 118 of Fig. 3, and may  
6 be performed in software, firmware, hardware, or a combination thereof. Fig. 10  
7 is described with additional reference to components in Fig. 3.

8 Initially, registration authority 118 receives the Get Certificate request (e.g.,  
9 an SCEP GetCert message) from router 210 (act 362). Registration authority 118  
10 decrypts the request (act 364), verifies the signature of the decrypted request (act  
11 366), and proceeds based on whether the signature is verified (act 368). If the  
12 signature cannot be successfully verified, then the message is dropped (act 370);  
13 registration authority 118 simply ignores the message. Alternatively, registration  
14 authority 118 may return an indication to router 210 that the signature could not be  
15 verified.

16 However, if the signature is successfully verified, then registration authority  
17 118 extracts the certificate serial number from the decrypted request (act 372).  
18 This serial number can be extracted by obtaining the serial number specified in the  
19 request (e.g., as the certificate serial number of the signing certificate of the  
20 request).

21 Registration authority 118 then uses the extracted serial number to generate  
22 a Get Certificate by Serial Number request (act 374). The Get Certificate by  
23 Serial Number request is then digitally signed and transmitted to certificate  
24 authority 116 (act 376), which in turn accesses its records to identify the certificate  
25

1 corresponding to the given serial number. This certificate is then returned by  
2 certificate authority 116 to registration authority 118 (act 378).

3 Registration authority 118 then generates an SCEP response message  
4 including the certificate received in act 378 (act 380). Registration authority 118  
5 then encrypts and digitally signs the SCEP response message including the  
6 certificate, and returns the encrypted and signed SCEP response message to router  
7 210 (act 382).

### 8 9 **Get CA Request**

10 Returning to Fig. 3, router 210 may also send a Get CA request as SCEP  
11 request 212. The request is an HTTP Get call to a URL hosted by registration  
12 authority 118. The URL is made available to router 210 during setup or  
13 configuration of router 210. Registration authority 118 responds to such a request  
14 by returning the requested certificates to router 210.

15 Fig. 11 is a flowchart illustrating an exemplary process for handling a Get  
16 Certificate Authority Certificate request in accordance with certain embodiments  
17 of the invention. The process of Fig. 11 is implemented by registration authority  
18 118 of Fig. 3, and may be performed in software, firmware, hardware, or a  
19 combination thereof. Fig. 11 is described with additional reference to components  
20 in Fig. 3.

21 Initially, a Get CA request is received by registration authority 118 from  
22 router 210 (act 400). Upon receipt of the request, registration authority 118  
23 obtains a DLL name identified by the request (act 402). In one implementation, an  
24 exemplary Get CA request from router 210 is in the following form:

1 GET mscep.dll/cgi-bin/pkiclient.exe?operation=GetCACert&message=  
2 <Base64 encoded authority issuer identifier>

3 In this implementation, registration authority 118 is implemented as an IIS  
4 (Internet Information Server) ISAPI (Internet Server Application Programming  
5 Interface) DLL. Upon receipt of such a request, IIS parses the input through to  
6 identify the first DLL and attempts to load that DLL if necessary. Thus, the  
7 remainder of the request can be ignored by registration authority 118 in  
8 determining how to respond to the request.

9 Registration authority 118 is the identified DLL, which in the illustrated  
10 example is "mscep.dll", and passes the request to response module 238 (act 404).  
11 In response to being passed the message (either in its entirety, or a part thereof),  
12 response module 238 generates a degenerated PKCS #7 message including the  
13 signing certificate and the encryption certificate of registration authority 118 (act  
14 406), and returns the degenerated PKCS #7 message to the router (act 408). Thus,  
15 router 210 requests the certificates for the certificate authority, but receives the  
16 certificates for the registration authority instead.

17 Alternatively, registration authority 118 may include a certificate chain in  
18 the message it generates in act 408. By way of example, MSCEP DLL 328 may  
19 send a certificate request to certificate authority 116, which returns the certificate  
20 of certificate authority 116 and a certificate chain that extends up to its root  
21 certificate.

## 22 **Password Handling**

23 Returning to Fig. 3, router 210 may also make use of a password to  
24 authenticate itself to certificate authority 116 (actually registration authority 118,  
25 but router 210 is not aware of this). The password allows registration authority

1 118 (and thus certificate authority 116, which trusts registration authority 118) to  
2 know that a particular request actually came from the router claiming to have sent  
3 it. The password may be used with one or more of the different types of SCEP  
4 requests 212 discussed above. By way of example, the password may be used  
5 with the router enrollment request.

6 Fig. 12 is a flowchart illustrating an exemplary process for distributing and  
7 verifying passwords in accordance with certain embodiments of the invention.  
8 The process of Fig. 12 is implemented by registration authority 118 of Fig. 3, and  
9 may be performed in software, firmware, hardware, or a combination thereof. Fig.  
10 12 is described with additional reference to components in Fig. 3.

11 Initially, registration authority 118 receives a request for a password (act  
12 430). This request is received via a mechanism that allows registration authority  
13 118 to authenticate the requestor, such as by use of SSL (Secure Sockets Layer) to  
14 authenticate the requestor when accessing web site 240 of Fig. 3. The requestor  
15 could be a computer being operated by a router administrator, or alternatively  
16 router 210. Upon receipt of the request, registration authority 118 attempts to  
17 authenticate the requestor, such as the router administrator, (act 432) and proceeds  
18 based on whether the authentication is successful (act 434). If the requestor  
19 cannot be authenticated, then the request for a password is denied (act 436). The  
20 request may simply be ignored, or alternatively an indication may be returned to  
21 the requestor that the request for a password is denied.

22 However, if the router is authenticated, then registration authority 118  
23 proceeds to generate a password and add the newly generated password to  
24 password table 226 (act 438). The password can be generated by registration  
25 authority 118 in any of a wide variety of conventional manners, such as by

1 generating a random (or pseudo-random) number and/or sequence of letters. The  
2 generated number may then be placed into a particular format if needed by either  
3 router 210 or certificate authority 116, such as hexadecimal format, binary coded  
4 decimal format, etc.

5 The password added to password table 226 is removed from table 226 after  
6 a period of time. In one implementation, each password in table 226 is kept in  
7 table 226 for sixty minutes and then removed. This period of time can optionally  
8 be configurable by an administrator.

9 Registration authority 118 then returns the newly generated password to  
10 requestor (act 440). This return of the password is done in a secure manner, such  
11 as by use of SSL.

12 Eventually, registration authority 118 receives a request from router 210  
13 that includes a password that needs to be verified (act 442). Upon receipt of such  
14 a request, registration authority 118 determines whether the received password is  
15 in password table 226 (act 444). If the received password is not in password table  
16 226, then the request is rejected (act 446). The request can simply be ignored, or  
17 alternatively a rejection response can be returned to router 210 (e.g., informing  
18 router 210 that the password it provided was not valid).

19 However, if the password is in password table 226, then the request is  
20 processed by registration authority 118 (act 448). Registration authority 118 may  
21 also optionally remove the password from password table 226 (act 450), thereby  
22 adding an additional level of security by allowing each password to be used only  
23 once.  
24  
25



1 **Conclusion**

2       Thus, a VPN enrollment protocol gateway has been described. The  
3 protocol gateway is implemented as a registration authority that is trusted by the  
4 certificate authority, and operates as an intermediary between the router and the  
5 certificate authority. The protocol gateway advantageously allows routers  
6 operating in accordance with one protocol to obtain and maintain certificates for a  
7 VPN from a certificate authority operating in accordance with another protocol.

8       Although the description above uses language that is specific to structural  
9 features and/or methodological acts, it is to be understood that the invention  
10 defined in the appended claims is not limited to the specific features or acts  
11 described. Rather, the specific features and acts are disclosed as exemplary forms  
12 of implementing the invention.  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25

## CLAIMS

1. A registration authority comprising:

a protocol converter coupled to receive messages from a router targeting a certificate authority, and to receive messages from the certificate authority targeting the router;

wherein the protocol converter is configured to convert the messages received from the router in accordance with a first protocol and convert the messages received from the router to a second protocol and subsequently communicate the converted messages to the certificate authority; and

wherein the protocol converter is further configured to convert the messages received from the certificate authority in accordance with the second protocol and convert the messages received from the certificate authority to the first protocol and subsequently communicate the converted messages to the router.

2. A registration authority as recited in claim 1, wherein the registration authority is independent of the certificate authority.

3. A registration authority as recited in claim 1, wherein the first protocol is a Simple Certificate Enrollment Protocol (SCEP) enrollment protocol.

4. A registration authority as recited in claim 1, wherein the second protocol is a Public-Key Cryptography Standards (PKCS) enrollment protocol.

1           5.     A registration authority as recited in claim 1, wherein the registration  
2 authority conforms to the network Working Group Request for Comments 2459  
3 standard.

4  
5           6.     A registration authority as recited in claim 1, wherein the messages  
6 received from the router comprise one or more of: a router enrollment message, a  
7 get certificate revocation list (CRL) message, a get certificate message, and a get  
8 certificate authority (CA) certificate message.

9  
10          7.     A registration authority as recited in claim 1, wherein each message  
11 received from the certificate authority comprises a response to a message received  
12 by the registration authority from the router.

13  
14          8.     A registration authority as recited in claim 1, wherein the router is  
15 unaware that it is communicating with a registration authority rather than directly  
16 with the certificate authority.

17  
18          9.     A registration authority as recited in claim 1, further comprising a  
19 transaction ID table configured to maintain a mapping of router transaction IDs  
20 received from the router to certificate authority request IDs received from the  
21 certificate authority.

1           **10.**    A registration authority as recited in claim 1, further comprising a  
2 request hash table configured to maintain a mapping of certificate authority  
3 request IDs to hash values of the router requests.

4  
5           **11.**    A registration authority as recited in claim 1, further comprising a  
6 password table configured to maintain a valid password issued to the router.

7  
8           **12.**    A registration authority as recited in claim 1, further comprising a  
9 module configured to receive a request for a certificate of the certificate authority  
10 and, in response to the request, return a certificate of the registration authority.

11  
12           **13.**    A registration authority as recited in claim 12, wherein the  
13 registration authority is a dynamically linked library.

14  
15           **14.**    One or more computer-readable media having stored thereon a  
16 computer program that, when executed by one or more processors of a computing  
17 device, causes the one or more processors to perform acts including:

18           transmitting a request for an enrollment certificate for a virtual private  
19 network to a registration authority operating independently of a certificate  
20 authority.

1           15. One or more computer-readable media as recited in claim 14,  
2 wherein the computer program further causes the one or more processors to  
3 transmit additional requests regarding maintaining enrollment in the virtual private  
4 network to the registration authority.

5  
6           16. One or more computer-readable media as recited in claim 14,  
7 wherein the computing device comprises a router.

8  
9           17. One or more computer-readable media having stored thereon a  
10 computer program that, when executed by one or more processors of a registration  
11 authority, causes the one or more processors to perform acts including:

12           receiving, from a device, a first message in accordance with a first protocol;  
13           generating, based on the first message, a second message in accordance  
14 with a second protocol;

15           sending the second message to a certificate authority;  
16           receiving, from the certificate authority, a third message in response to the  
17 second message and in accordance with the second protocol;

18           generating, based on the third message, a fourth message in accordance  
19 with the first protocol; and

20           sending the fourth message to the device as a response to the first message.

21  
22           18. One or more computer readable media as recited in claim 17,  
23 wherein the device comprises a router.

1           **19.** One or more computer-readable media as recited in claim 17,  
2 wherein the first message comprises an enrollment message.

3  
4           **20.** One or more computer-readable media as recited in claim 19,  
5 wherein generating the second message comprises:

6           verifying that the first message has been digitally signed by the device;  
7           decrypting the first message;  
8           extracting a certificate enrollment request from the first message;  
9           generating a certificate authority request including the certificate  
10 enrollment request and a subject alternative names extension; and  
11           creating the second message by digitally signing the certificate authority  
12 request.

13  
14           **21.** One or more computer-readable media as recited in claim 19,  
15 wherein generating the fourth message comprises:

16           extracting a certificate from the third message;  
17           generating a response including the certificate;  
18           encrypting the response; and  
19           creating the fourth message by digitally signing the encrypted response.

20  
21           **22.** One or more computer-readable media as recited in claim 21,  
22 wherein extracting the certificate comprises accessing a set of certificates  
23 corresponding to the third message.  
24  
25

1           **23.**   One or more computer-readable media as recited in claim 21,  
2 wherein the computer program further causes the one or more processors to  
3 perform acts including:

4           extracting a certificate chain from the third message; and  
5           including the certificate chain in the response.  
6

7           **24.**   One or more computer-readable media as recited in claim 19,  
8 wherein the third message comprises a certificate authority pending response.  
9

10          **25.**   One or more computer-readable media as recited in claim 24,  
11 wherein generating the fourth message comprises:

12          generating a pending response;  
13          encrypting the pending response; and  
14          creating the fourth message by digitally signing the encrypted pending  
15 response.  
16

17          **26.**   One or more computer-readable media as recited in claim 24,  
18 wherein the computer program further causes the one or more processors to  
19 perform acts, in response to the certificate authority pending response, generating:

20          a hash value based on the enrollment message;  
21          a hash table entry mapping a pending response ID, corresponding to the  
22 certificate authority pending response, to the hash value; and  
23          a transaction ID table entry mapping the transaction ID, corresponding to  
24 the enrollment message, to a pending response ID corresponding to the certificate  
25 authority pending response.

1  
2       27. One or more computer-readable media as recited in claim 26,  
3 wherein the computer program further causes the one or more processors to  
4 perform acts including:

5       receiving an additional enrollment message from the device;  
6       accessing the transaction ID table to obtain the pending response ID  
7 corresponding to the additional enrollment message; and  
8       transmitting, to the certificate authority, a certificate request including the  
9 pending response ID.

10  
11       28. One or more computer-readable media as recited in claim 26,  
12 wherein the computer program further causes the one or more processors to  
13 perform acts including:

14       receiving an additional enrollment message from the device;  
15       generating a new hash value based on the additional enrollment message;  
16       checking whether an entry in the hash table matches the new hash value;  
17 and  
18       if an entry in the hash table matches the new hash value, then,  
19       obtaining a pending response ID, from the hash table, corresponding  
20 to the new hash value, and  
21       transmitting, to the certificate authority, a certificate request  
22 including the pending response ID.



1           **29.** One or more computer-readable media as recited in claim 26,  
2 wherein the computer program further causes the one or more processors to  
3 perform acts including:

4           maintaining the hash table entry in the hash table for a selected amount of  
5 time.

6  
7           **30.** One or more computer-readable media as recited in claim 26,  
8 wherein the computer program further causes the one or more processors to  
9 perform acts including:

10           maintaining the transaction ID table entry in the transaction ID table for a  
11 selected amount of time.

12  
13           **31.** One or more computer-readable media as recited in claim 17,  
14 wherein the first message comprises a get certificate revocation list (CRL)  
15 message.

16  
17           **32.** One or more computer-readable media as recited in claim 31,  
18 wherein generating the second message comprises:

19           decrypting the first message;

20           verifying that the first message has been digitally signed by the device;

21           extracting a certificate serial number from the decrypted first message; and

22           creating, as the second message, a get certificate by serial number request.  
23  
24  
25

1           **33.** One or more computer-readable media as recited in claim 31,  
2 wherein generating the fourth message comprises:

3           extracting a certificate from the third message;  
4           extracting a certificate revocation list distribution point from the certificate;  
5           obtaining a certificate revocation list based on the certificate revocation list  
6 distribution point; and

7           generating, as the fourth message, a response including the certificate  
8 revocation list.

9  
10           **34.** One or more computer-readable media as recited in claim 33,  
11 wherein the certificate revocation list distribution point comprises a uniform  
12 resource locator (URL).

13  
14           **35.** One or more computer-readable media as recited in claim 33,  
15 wherein obtaining the certificate revocation list further comprises retrieving the  
16 certificate revocation list from the certificate revocation list distribution point.

17  
18           **36.** One or more computer-readable media as recited in claim 17,  
19 wherein the first message comprises a get certificate message.

20  
21           **37.** One or more computer-readable media as recited in claim 36,  
22 wherein generating the second message comprises:

23           decrypting the first message;  
24           verifying that the first message has been digitally signed by the device;  
25           extracting a certificate serial number from the decrypted first message; and

1 creating, as the second message, a get certificate by serial number request.  
2

3 **38.** One or more computer-readable media as recited in claim 17,  
4 wherein generating the fourth message comprises:  
5 extracting a certificate from the third message; and  
6 generating, as the fourth message, a response including the certificate.  
7

8 **39.** One or more computer-readable media as recited in claim 38,  
9 wherein generating the fourth message further comprises:  
10 extracting a certificate chain from the third message; and  
11 including the certificate chain in the response.  
12

13 **40.** A method implemented at a registration authority, the method  
14 comprising:  
15 receiving, from a device, a get certificate authority certificate request;  
16 generating a response including a certificate of the registration authority;  
17 and  
18 returning the response to the device.  
19

20 **41.** A method as recited in claim 40, wherein the device comprises a  
21 router.  
22  
23  
24  
25

1           **42.**     A method as recited in claim 40, wherein the get certificate authority  
2 certificate request identifies a dynamically linked library (DLL) that is the  
3 registration authority.

4  
5           **43.**     A method as recited in claim 40, wherein the response comprises a  
6 degenerated message.

7  
8           **44.**     A method as recited in claim 40, wherein the response includes both  
9 a signing certificate of the registration authority and an encryption certificate of  
10 the registration authority.

11  
12           **45.**     A method as recited in claim 40, wherein the response further  
13 includes a certificate chain of the certificate authority.

14  
15           **46.**     One or more computer-readable memories containing a computer  
16 program that is executable by a processor to perform the method recited in claim  
17 40.

18  
19           **47.**     A method comprising:  
20         receiving a request, from a requestor, for a password to be used by a device  
21 when communicating with a registration authority;  
22         authenticating the requestor;  
23         generating the password;  
24         adding the password to a password table; and  
25         returning the password to the requestor for use by the device.

1  
2       **48.**     A method as recited in claim 47, wherein the device comprises a  
3 router.

4  
5       **49.**     A method as recited in claim 47, wherein generating the password  
6 comprises generating a random number as the password.

7  
8       **50.**     A method as recited in claim 47, wherein receiving, authenticating,  
9 and returning include using Secure Sockets Layer (SSL) to maintain secure  
10 communication with the device.

11  
12       **51.**     A method as recited in claim 47, further comprising keeping the  
13 password active for a selected amount of time.

14  
15       **52.**     A method as recited in claim 51, wherein keeping the password  
16 active for a selected amount of time comprises marking the password as invalid  
17 after the selected amount of time.

18  
19       **53.**     A method as recited in claim 51, wherein keeping the password  
20 active for a selected amount of time comprises removing the password from the  
21 password table after the selected amount of time.

22  
23       **54.**     A method as recited in claim 47, further comprising:  
24       receiving a request from the device, the request including a request  
25 password;

1 checking whether the request password is included in the password table;  
2 and  
3 processing the request if the request password is included in the password  
4 table, otherwise rejecting the request.  
5

6 **55.** A method as recited in claim 54, further comprising removing, if the  
7 request password is included in the password table, the request password from the  
8 password table.  
9

10 **56.** One or more computer-readable memories containing a computer  
11 program that is executable by a processor to perform the method recited in claim  
12 47.  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25

1 **ABSTRACT**

2 A virtual private network (VPN) enrollment protocol gateway is described  
3 herein. The protocol gateway is implemented as a registration authority that  
4 operates as an intermediary between routers and a certificate authority, allowing  
5 routers operating in accordance with one protocol to obtain and maintain  
6 certificates for a VPN from a certificate authority operating in accordance with  
7 another protocol. In accordance with one aspect, the gateway protocol supports  
8 various requests from the router, including router enrollment requests, get  
9 certificate revocation list request, get certificate requests, get certificate authority  
10 certificate requests, and password requests.

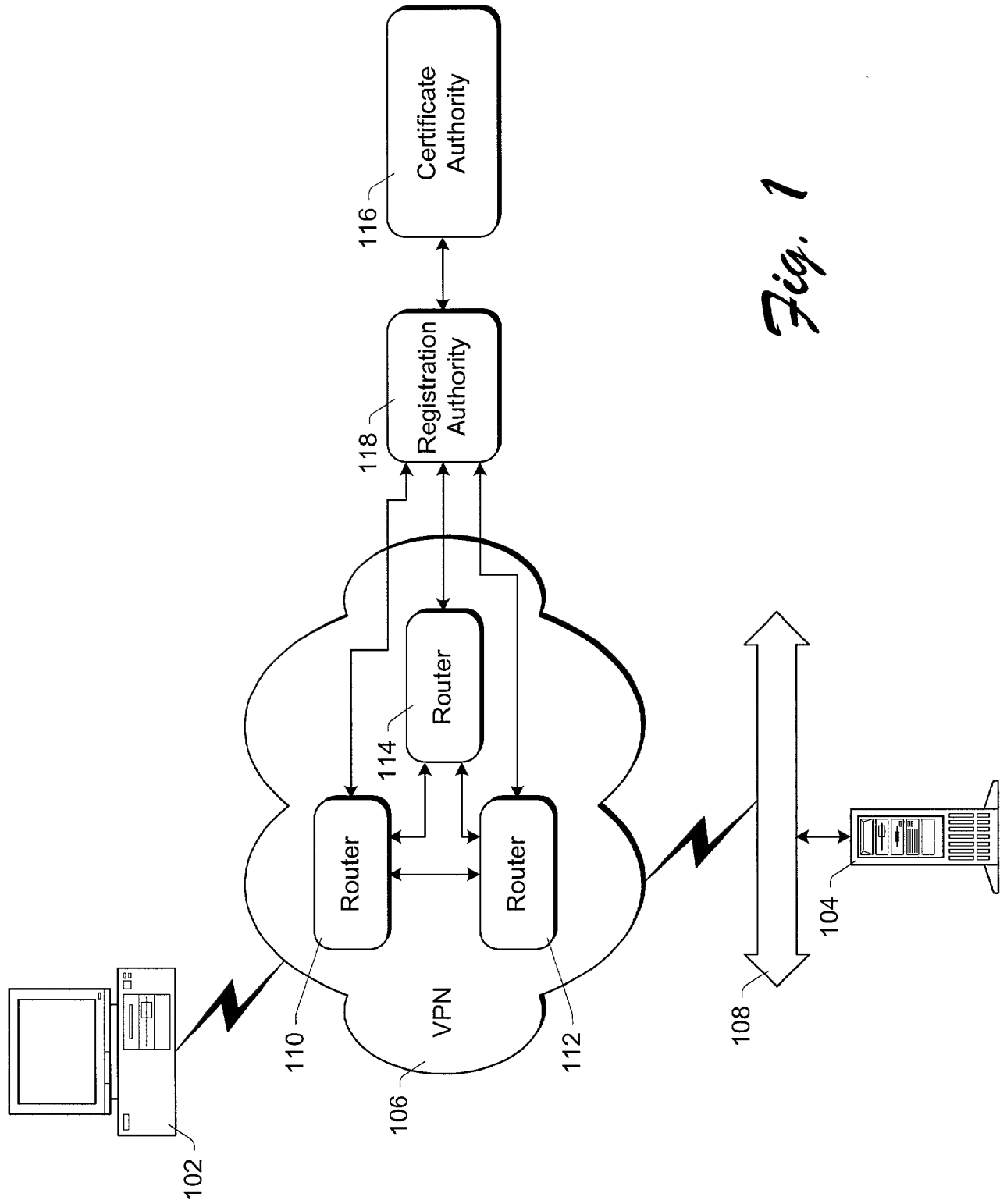
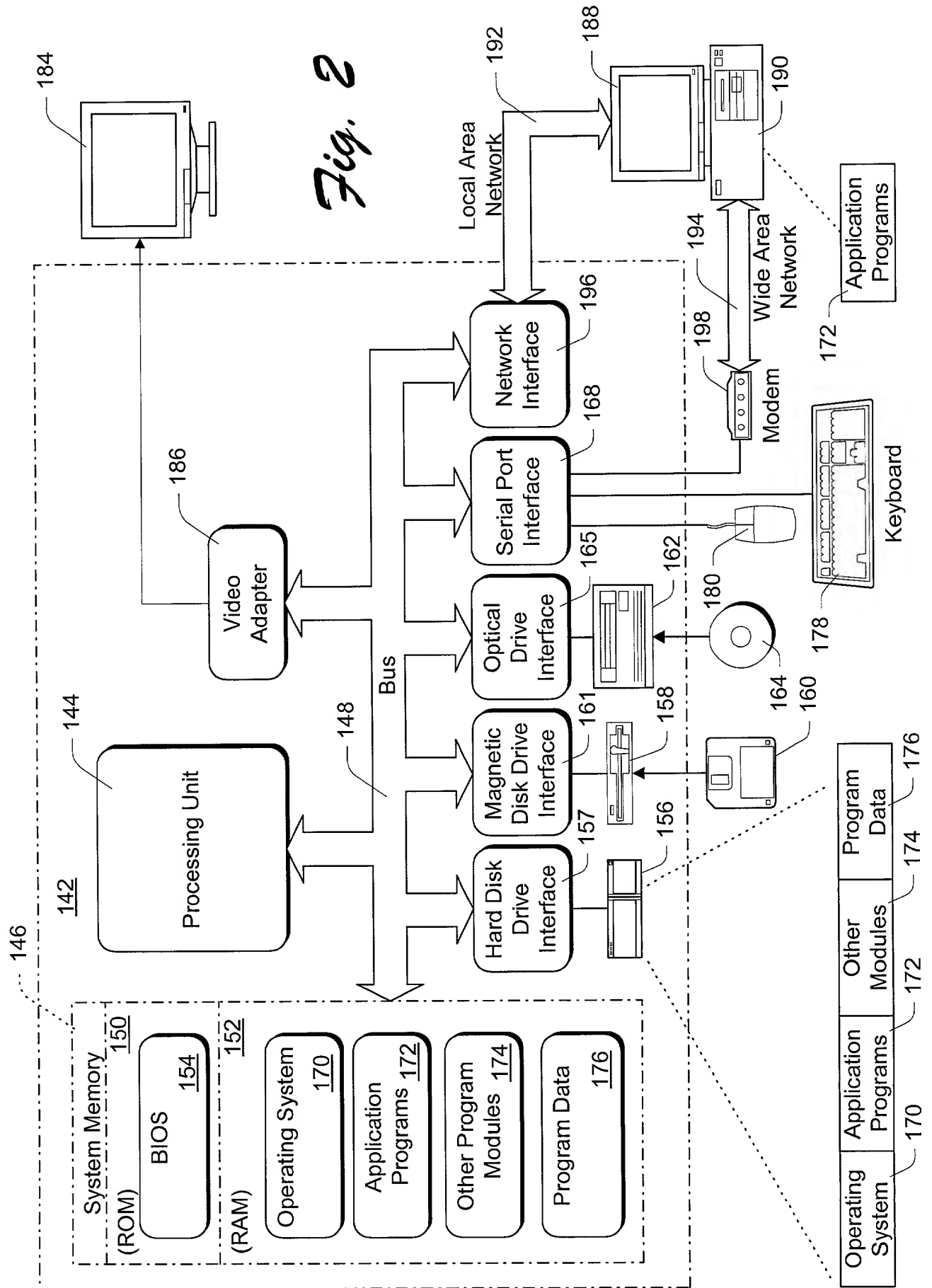


Fig. 1





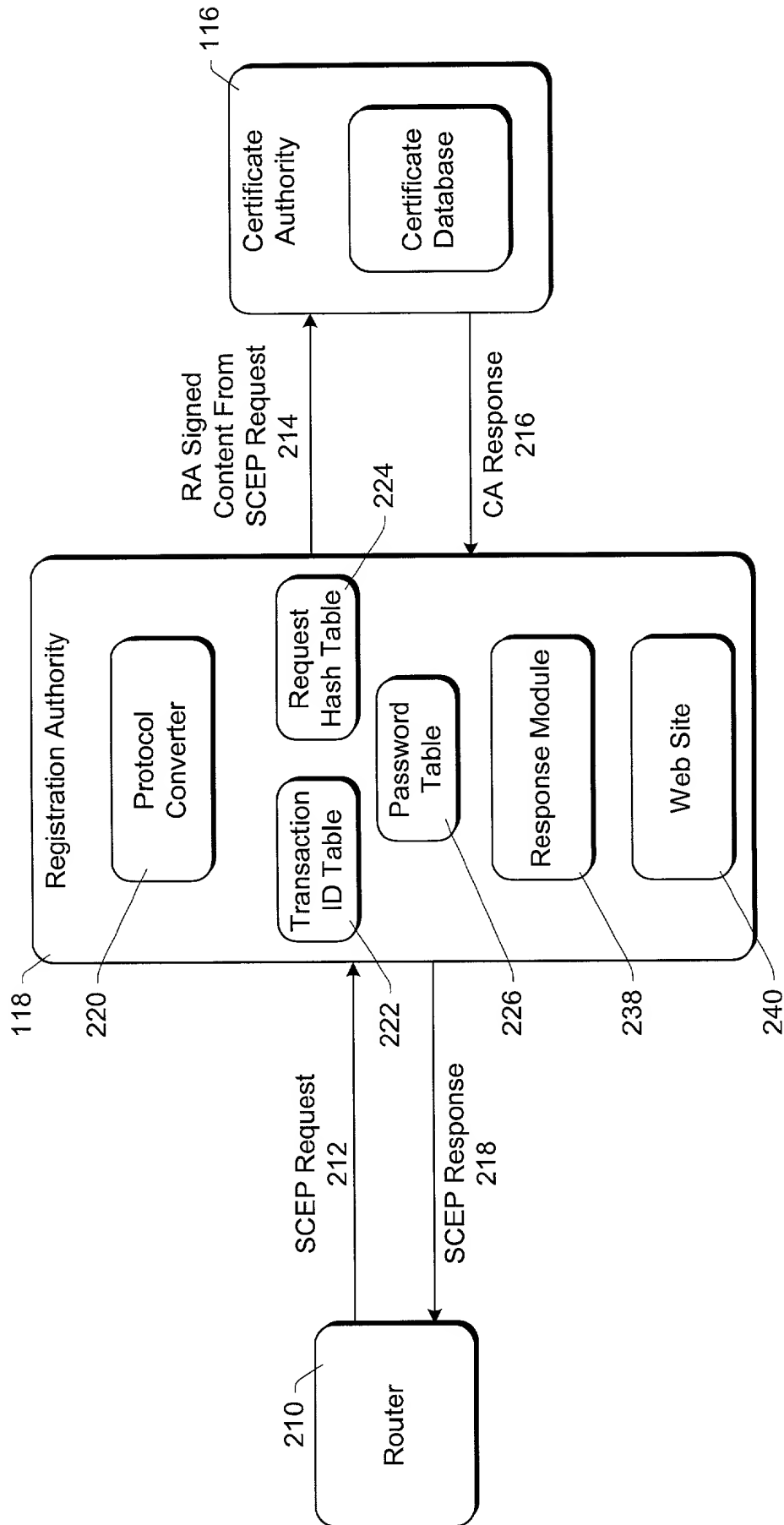


Fig. 3

222 ↘

228 ↘	230 ↘
Router Transaction ID (1)	Certificate Authority Request ID (1)
Router Transaction ID (2)	Certificate Authority Request ID (2)
⋮	⋮
Router Transaction ID (n)	Certificate Authority Request ID (n)

Fig. 4

224 ↘

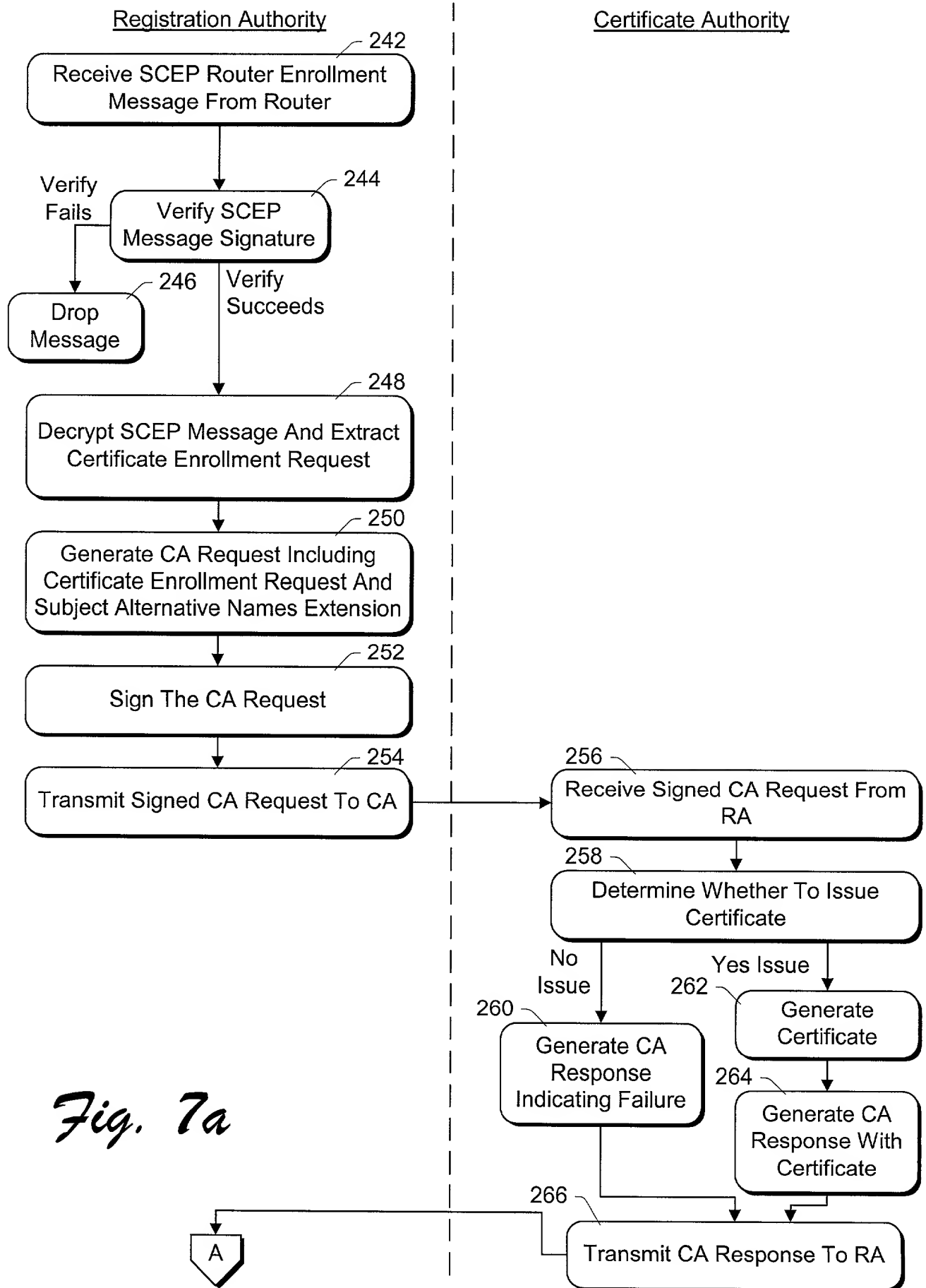
232 ↘	234 ↘
Certificate Authority Request ID (1)	Hash of Request (1)
Certificate Authority Request ID (2)	Hash of Request (2)
⋮	⋮
Certificate Authority Request ID (m)	Hash of Request (m)

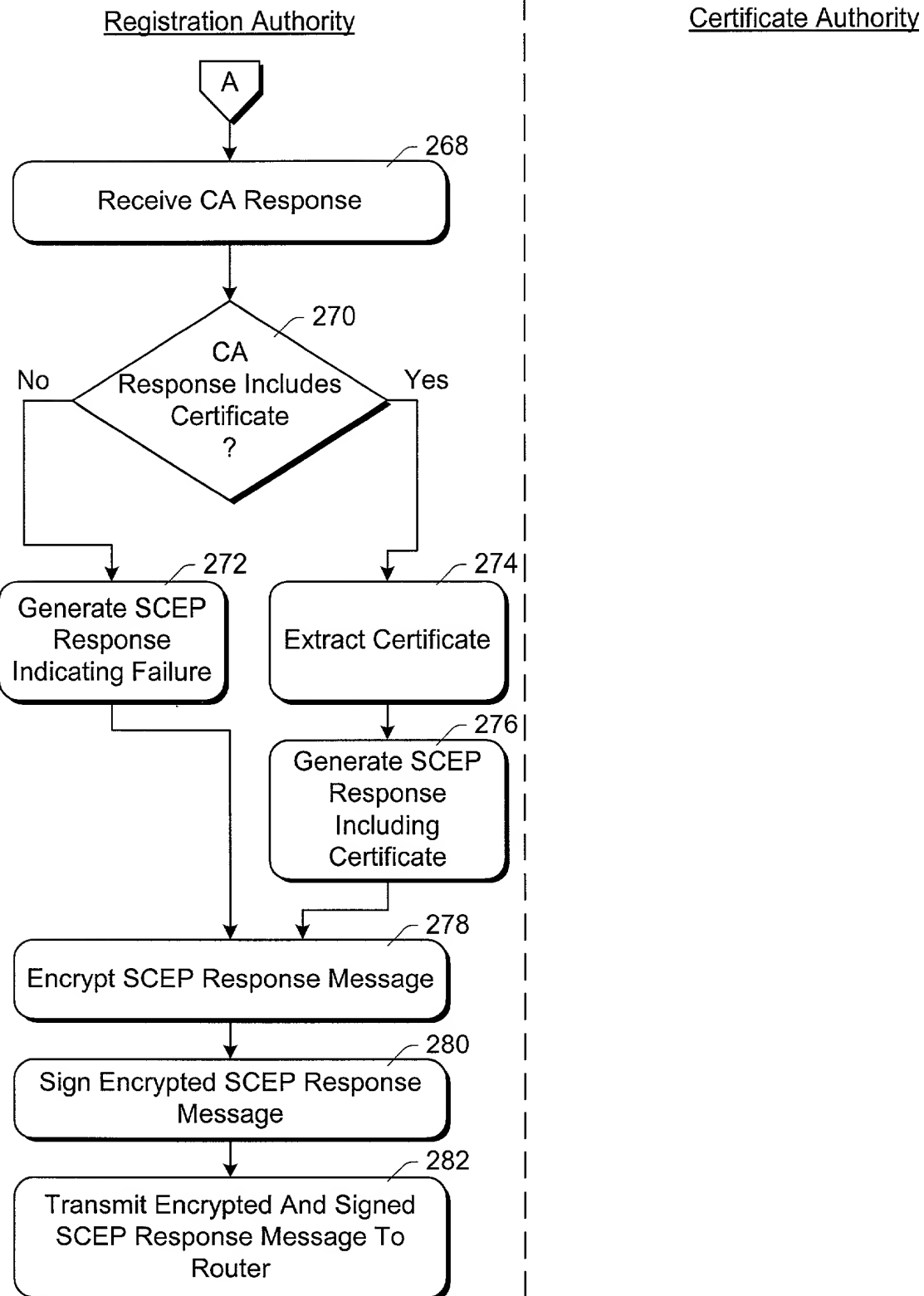
Fig. 5

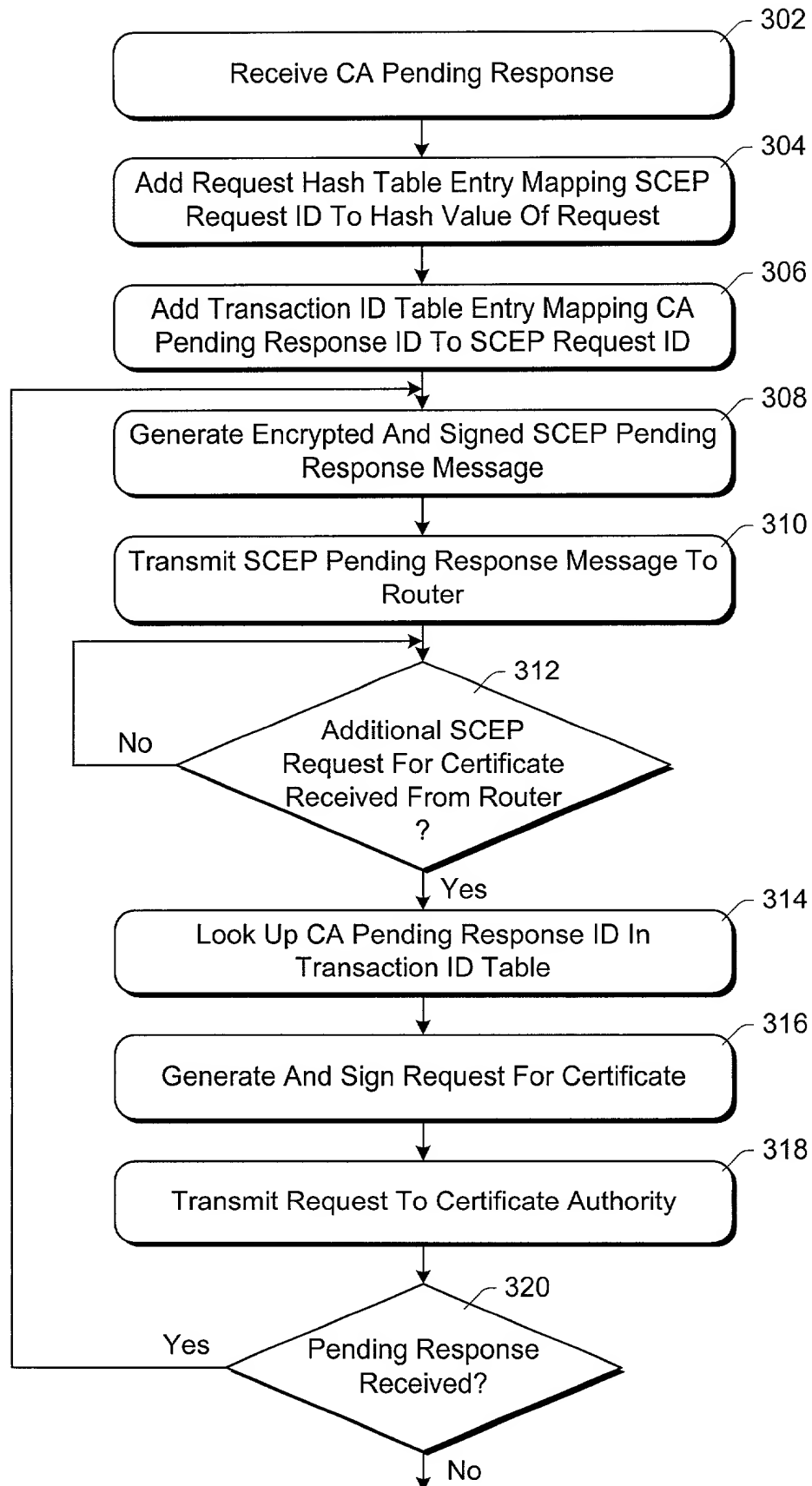
224 ↘

236 ↘
Password (1)
Password (2)
⋮
Password (x)

Fig. 6



*Fig. 76*

*Fig. 8*

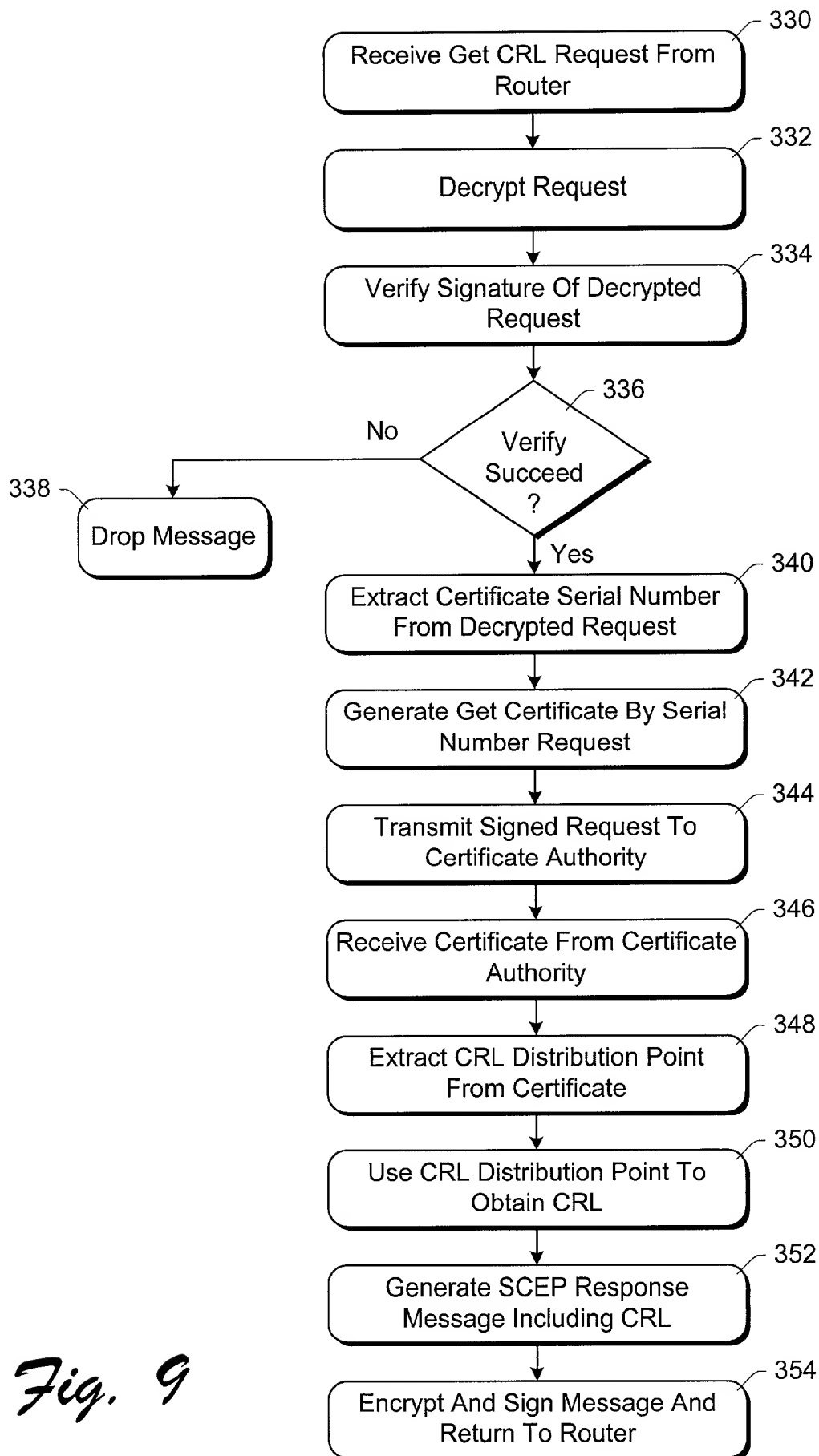
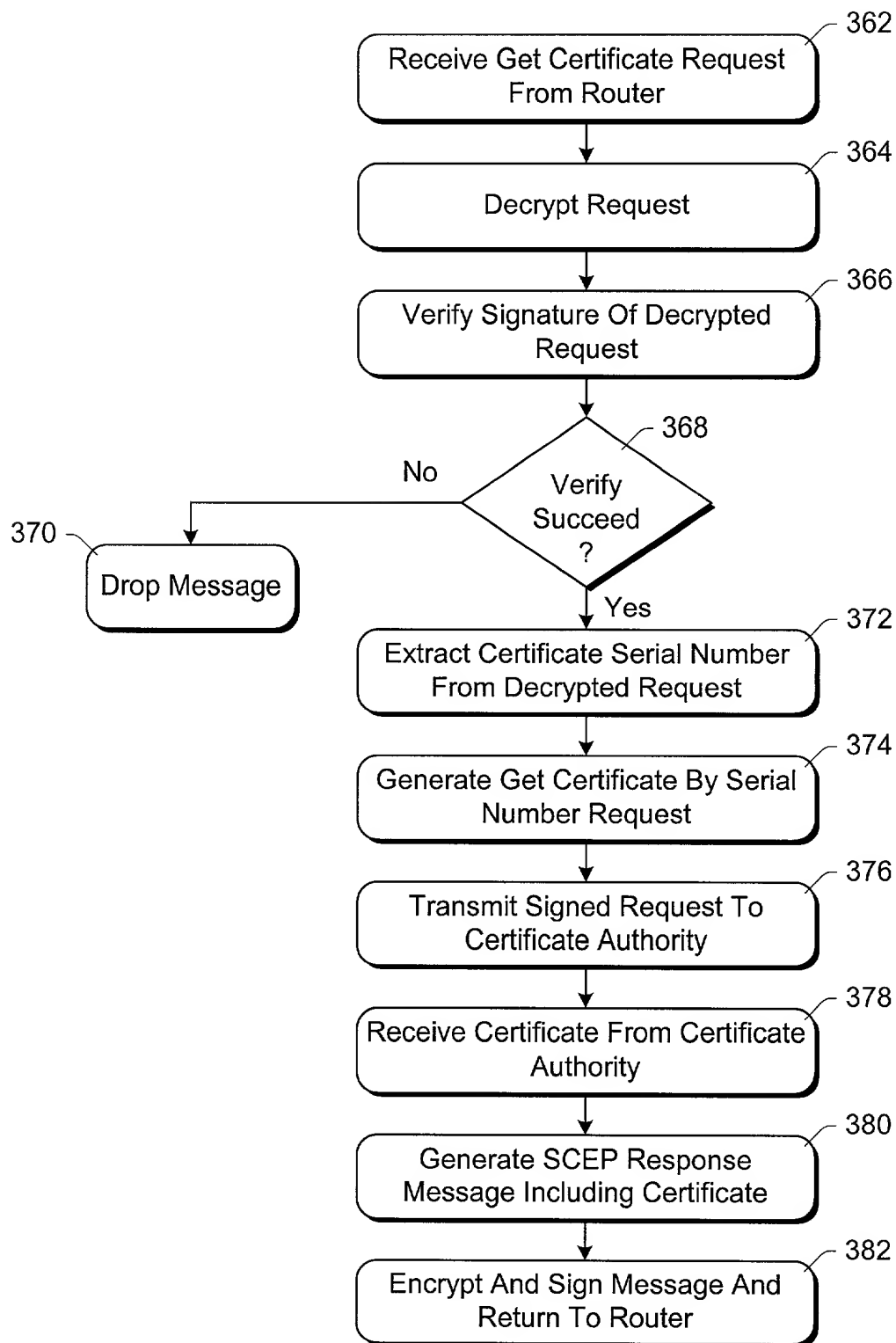
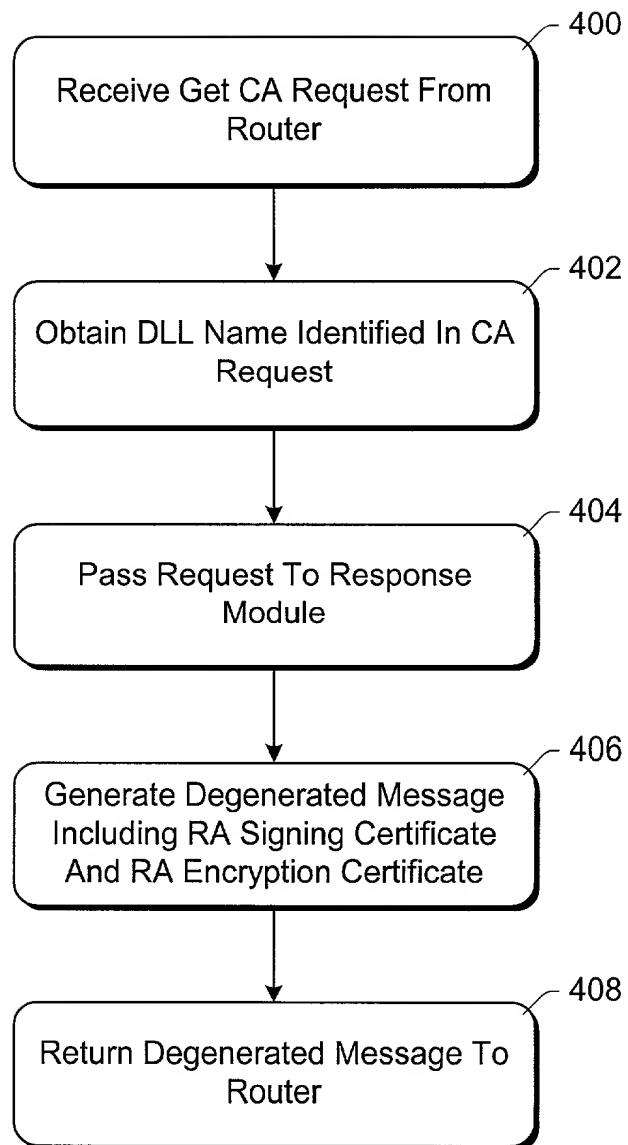
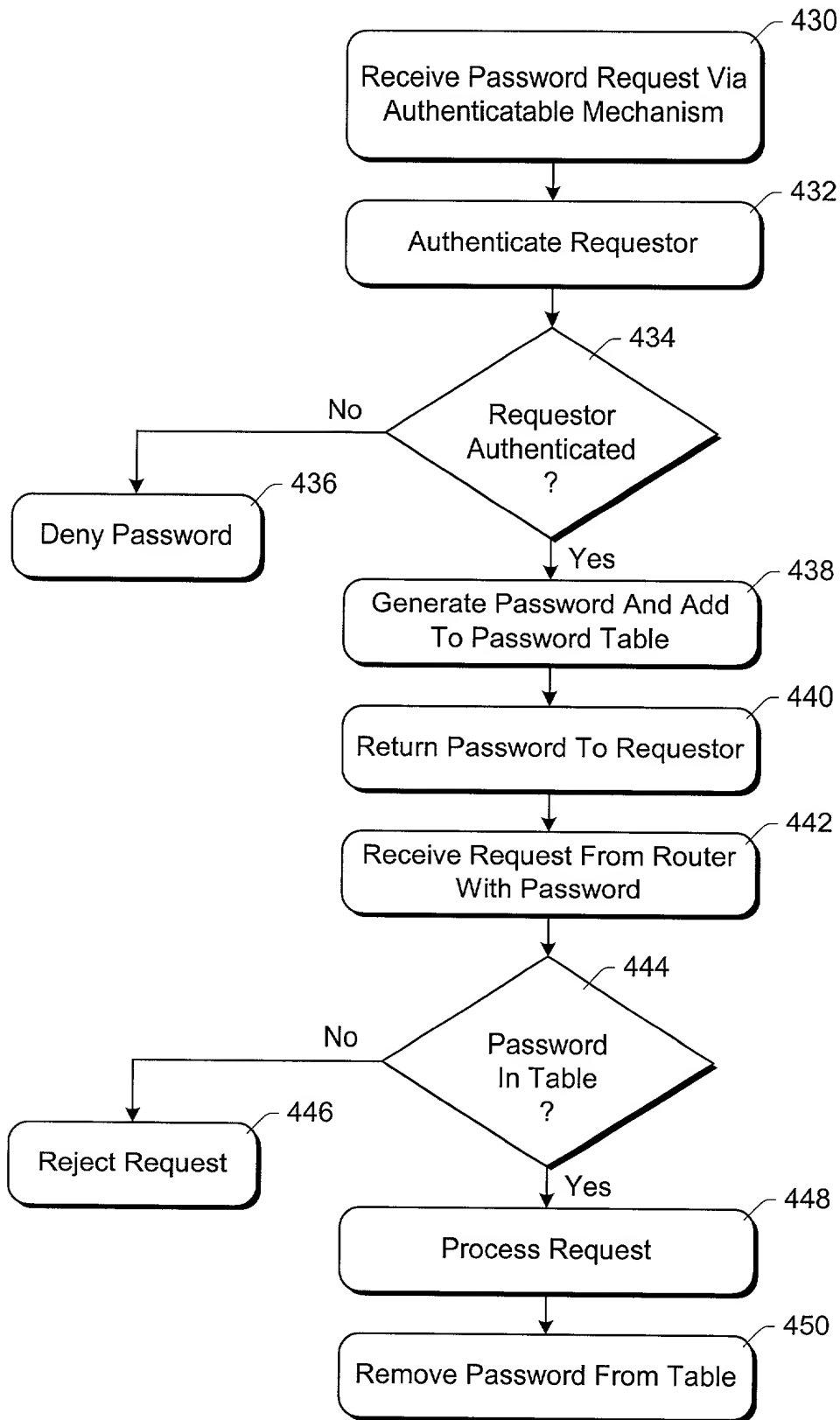


Fig. 9

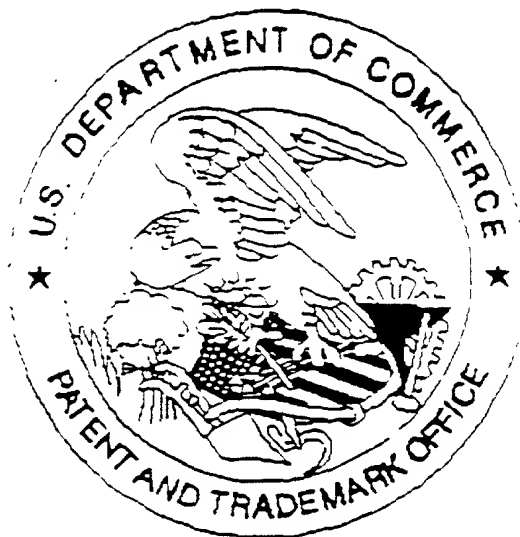
*Fig. 10*



*Fig. 11*

*Fig. 12*

United States Patent & Trademark Office  
Office of Initial Patent Examination -- Scanning Division



Application deficiencies were found during scanning:

☐ Page(s) \_\_\_\_\_ of \_\_\_\_\_ were not present  
for scanning. (Document title)

☐ Page(s) \_\_\_\_\_ of \_\_\_\_\_ were not present  
for scanning. (Document title)

NO LARGE ENTITY STATUS

☐ Scanned copy is best available.